



Os objetivos específicos deste capítulo são:

- Fornecer as diretrizes básicas para a confecção de algoritmos legíveis, bem documentados, claros e de fácil compreensão.
- Agilizar o desenvolvimento crítico dos alunos para a elaboração de algoritmos através de exemplos de algoritmos com boa qualidade.
- Executar a confecção de fluxogramas tradicionais e estruturados e a sua conversão para o PORTUGOL, a vice-versa.

Neste capítulo são apresentadas "máximas de programação" que são regras práticas para a elaboração de algoritmos com qualidade. Alguns exemplos e contra-exemplos são apresentados. Em seguida são propostos diversos problemas e analisados os dados para a confecção dos respectivos algoritmos. Os exemplos serão apresentados utilizando alternadamente fluxogramas (estruturados e tradicionais) e PORTUGOL. Também são propostos e resolvidos exercícios de conversão de PORTUGOL para fluxogramas estruturados ou não, e como fazer a conversão. Finalmente, dando um fluxograma, será analisado se ele é um fluxograma estruturado ou não, e como fazer a conversão.

3.1 MÁXIMAS DA PROGRAMAÇÃO

1. *Algoritmos devem ser feitos para serem lidos por seres humanos.* Tenha em mente que seus algoritmos deverão ser lidos e entendidos por outras pessoas (e por você mesmo) de tal forma que possam ser corrigidos, receber manutenção e ser modificados.

2. *Escreva os comentários no momento em que estiver escrevendo o algoritmo.* Um algoritmo não documentado é um dos maiores erros que um programador pode cometer e é sinal de amadorismo (mesmo com 10 anos de experiência). Como o objetivo de se escrever comentários é facilitar o entendimento do algoritmo, eles devem ser tão bem conhecidos quanto o próprio algoritmo. E a melhor maneira de se conseguir isso é escrevê-lo nos momentos de maior intimidade com os detalhes, ou seja, durante a resolução do problema.

3. *Os comentários deverão acrescentar alguma coisa; não apenas para frisar os comandos.* O conjunto de comandos nos diz o que está sendo feito; os comentários deverão nos dizer por quê.

4. *Use comentários no prólogo.* Todo algoritmo ou procedimento deverá ter comentários no seu prólogo para explicar o que ele faz e fornecer instruções para seu uso. Alguns destes comentários seriam:

- a) Uma descrição do que faz o algoritmo.
- b) Como utilizá-lo.
- c) Explicação do significado das variáveis mais importantes.
- d) Estruturas de dados utilizadas.
- e) Os nomes de quaisquer métodos especiais utilizados, juntamente com referências nas quais mais informações possam ser encontradas.
- f) Autor.
- g) Data de escrita.

5. *Utilize espaços em branco para melhorar a legibilidade.* Espaços em branco, inclusive linhas em branco, são valiosíssimos para melhorar a aparência de um algoritmo. Exemplo:

- a) Deixar uma linha em branco entre as declarações e o corpo do algoritmo.
- b) Deixar uma linha em branco antes e outra depois de um comentário.
- c) Separar grupos de comandos que executam funções lógicas distintas por uma ou mais linhas em branco.
- d) Utilizar brancos para indicar precedência de operadores. Ao invés de $A * B + C$ é bem mais agradável a forma $A * (B + C)$.

6. *Escolha nomes representativos para suas variáveis.* Os nomes das variáveis deverão identificar, o melhor possível, as quantidades que elas representam. Por exemplo, $X = Y + Z$ é muito menos claro que $PREÇO = CUSTO + LUCRO$. Uma seleção adequada de nomes de variáveis é o princípio mais importante da legibilidade de algoritmos.

7. *Um comando por linha é suficiente.* A utilização de vários comandos por linha é prejudicial por várias razões, dentre as quais destacam-se: